

ウェブサイト間における書評の相互交換を 目的とした文書構造の定義

A Definition of the Document Structure for the Online Exchange
of Book Reviews Published on Internet Sites

芦 田 昌 也
Ashida, Masaya

ABSTRACT

Collecting information related to a particular field or subject is important, as the accumulated information can give rise to collective intelligence. A book review provides information on the book, the author's argument and the reader's interpretation. A collection of book reviews also has the potential to yield collective intelligence. In order to make a aggregator site collecting book reviews on the Internet, it is necessary to transfer the book reviews as electronic documents from the websites that own the book reviews to the aggregator site. XML (Extensible Markup Language) is used for exchanging electronic documents over the Internet. This article defines the document structure for book reviews. The definition is written in XML Schema, one of the description languages for document structure.

1. はじめに

新聞や雑誌には、新刊書や話題書の書評を定期的に掲載するものがある。一部の書店では販売促進や来店者の知的好奇心を刺激することなどを目的として、該当書籍の販売箇所に短い書評を掲示している。インターネットによる情報発信が容易になった近年では、書店のサイトや個人のブログなどにも書評が掲載されている。

書籍の内容を評価する観点は、一般に評者によって異なる。異なる観点からの書評が集まることにより、当該書籍の様々な価値の発見が期待できる。このような情報の集約により、新たな価値や知識が創出される可能性があることはこれまでに指摘されている。既にオンライン書店などでは、感想や内容のレベルなど広義の書評が読者を通じて集約され、購入を検討している潜在的な読者への参考情報となっている。また、一つの作品に対する意見交換や解釈についての議論などを行い、読書体験を共有しようとするソーシャルリーディングという活動も始められている。このような情報の集約・蓄積・提供は今後も継続・進展することが予想される。現状では、情報の集約・蓄積・提供は、主にそれぞれのウェブサイト内で行われている。より活発な活動のためには、ウェブサイト間での情報の相互提供を容易にすることが望まれる。

そこで本稿では、インターネットを利用した情報の相互提供による書評の集約を実現するためのひとつのアプローチを検討する。まず、情報を集約するための方法をスタンドアロン型、ポータル型、コラボレーション型、マッシュアップ型、ハイブリッド型に類型化し、ウェブサイト間での書評の相互提供を実現するものとしてマッシュアップ型を提案する。次に、書評を機械可読形式の文書として記述するための方法として、XML (Extensible Markup Language) の使用を提案し、その文書構造を定義する。定義された文書構造を拡張し、実際の書評サイトで利用する文書構造の例を示す。最後に今後の展開と残された課題について述べる。

2. 情報集約の類型

情報を集約する形態に着目し、集約した情報を提供する集約側サイトの分類を示す。書評という情報の性質を考慮し、相互提供の実現に適すると考えられる類型を提示する。

2.1. スタンドアロン型

ウェブサイトがハイパーリンクも含めて他のウェブサイトやページで提供さ

れている情報を引用することなく、独自の情報を提供する形態を指すものとする。他のウェブサイト等からの情報を引用しないという点で情報提供の起点とみなすことができる。当該サイトで提供するすべての情報のコンテンツを保有し、情報の品質、表現の方法（デザインやレイアウト）を完全に制御できる。

2.2. ポータル型

コンテンツを有するウェブサイトやページの URL を保有することで情報の集約に換え、主にハイパーリンクにより情報を提供する形態を指すものとする。ディレクトリ型検索エンジンのように、人手による情報の選別が可能である。ハイパーリンクとして集約した情報のコンテンツそのものを保有しない。ハイパーリンクによる他のウェブサイトやウェブページの参照は、ウェブのひとつの利点であるが、URL の変更や消滅に対して脆弱である。情報の更新はコンテンツの所有者に任される。集約された情報は、そのコンテンツを保有するウェブサイトやウェブページに移動したのち、そこでのデザインに従って表示されるため、集約側サイトのデザインやレイアウトでコンテンツを配置することは困難である。

2.3. コラボレーション型

Wikipedia に代表されるように、特定・不特定、および多数・少数の情報提供者（以下、参加者）があらかじめ定められた集約側サイトにアクセスし、共同で情報の集約を図ると同時に集約した情報を提供する形態を指すものとする。参加者はひとつの集約側サイトを共有するから、互いに参加者により提供された情報を把握することができる。これにより参加者相互で不足する情報や誤った情報を追記・修正しやすいことに特徴がある。客観的事実や客観的知識を集約する場合には、参加者が相互に補完しあうことにより、効率的な情報の集約ができる。

一方、参加者による自由な編集が可能であることから、集約側サイトの趣旨を理解しない参加者によって意図せぬ変更が行われることもある。意図せぬ変更を防止するために、参加者を登録制にすることが可能である。そのような運

用形態では、集約側サイト管理者の負担が高くなることが予想される。

2.4. マッシュアップ型

コンテンツを保有するウェブサイトからコンテンツの提供を受けて情報を集約し、集約側サイトで情報を整理して提供する形態を指すものとする。コンテンツを保有するサイトからの提供が前提である。コンテンツの授受がプログラムを介して自動的に行われる場合、他者による意図せぬ変更が生じる可能性は低い。入手したコンテンツを集約側サイトで自由に配置することが可能である。コンテンツを提供しているサイトの検索は、ポータル型で有益な情報を掲載したサイトやウェブページの URL の検索と同程度の負担がある。現状では、コンテンツの再利用を前提として提供を行っているサイトはまだ少ない。

2.5. ハイブリッド型

スタンドアロン型、ポータル型、コラボレーション型、マッシュアップ型の2つ以上を組み合わせる情報を集約し提供する形態を示すものとする。オーソドックスなウェブサイトは、独自に提供する情報と関連する情報のハイパーリンクを保有するスタンドアロン型とポータル型のハイブリッド型である。

2.6. 書評の集約

書評は、対象とする書籍の内容について、主に評者自身の主張を述べるものである。記載内容は頻繁に更新されるものではない。また、ソーシャルリーディングのように、参加者相互での共同作業を前提とするものでなければ、記載内容の誤りや不足を他者が追記・修正する性質のものではない。さらに、他者により評者の意図に反した変更は認められない。

スタンドアロン型はすべての情報を制御できるため、内容の意図せぬ変更を防止することが比較的容易である。一方で、他の書評サイトからの引用を含まず、独自に書評を追加するにとどまるため、書評を集約する役割は十分ではない。

コラボレーション型では、誰もが自由に書評を投稿・閲覧できる利点がある。そのため集約側サイトが積極的に働き掛けることなく、書評を収集できる可能

性がある。集約側サイトには投稿された書評そのものがコンテンツとして蓄積され、そのサイト自体が価値を有するものとして捉えられる。一方で、書評の品質は投稿者次第であり集約側サイトでの選別は容易ではない。また、評者以外の参加者による内容変更を防止するために、参加者登録の実施、編集機能の制限、編集権限の設定などの運用を行うと、自由に投稿できる利点を失うと同時に、集約側サイトの管理コストが増加する。

ポータル型では、所有するコンテンツが URL であるため、集約側サイトにおいて書評そのものの内容の変更などは行われない。すでに記述された書評から選別できるため、集約側サイトの品質を保つことが可能である。その一方で、集約側サイトの統一されたデザインやレイアウトで書評を提供できないことから、集約側サイトからの情報提供という印象を利用者に与えることが難しい。

これらのことから、書評を保有するウェブサイトからのコンテンツの提供を前提にすると、マッシュアップ型のウェブサイトにより書評を集約することが最も適切であると考えられる。スタンドアロン型と組み合わせたハイブリッド型のウェブサイトにすることにより、そのウェブサイトの独自性を表現することができる。構築するウェブサイトの趣旨に沿って提供する書評を選別して収集することが可能である。特定分野の書籍についての書評を集積することも可能である。

2.7. 書評の提供

ウェブで閲覧できる書評の文書構造は類似するものが多い。文書構造が類似していることから、ウェブで公開されている書評には、書評としての標準的な項目がたいてい含まれる。このことは、書評をウェブサイト間で相互に提供することを容易にする。

ウェブサイト間で情報を授受する方法に Web API がある。コンテンツを保有するウェブサイトが、そのコンテンツを提供するための仕組みのひとつである。ウェブサイトが提供可能なコンテンツは Web API の仕様で定める。クライアントが Web API を利用して定められた形式の要求を送信すると、その要

求に応じてウェブサーバからコンテンツが返される。これらの処理はプログラムを介して自動的に行われる。

このような Web API を利用し、書評を保有する各ウェブサイトが相互にそのコンテンツを提供することにより、書評の集約を活発に進めることが可能になる。

3. 書評記述のための文書構造

Web API を利用してコンテンツを授受する代表的な形式のひとつに XML 形式がある。XML は文書の構造を記述し、デザインやレイアウトの情報は原則として含まない。そのため、コンテンツを提供するウェブサイトのデザインやレイアウトの影響をうけることなく、集約側サイトでコンテンツを自由に配置できる。また、独自の書評を蓄積する形式としても利用できる。XML で記述された書評を入手したウェブサイトにおいて再利用するには、文書構造の定義が明確でなければならない。そのために、Web API で書評を提供するウェブサイトは、その文書構造を定める必要がある。文書構造は XML のスキーマを定義する言語のひとつである XML Schema を用いて定義する。

文書構造が類似していることに着目し、標準的な項目はどのウェブサイトでも再利用可能な形式で定義し、各ウェブサイトで独自の拡張ができるよう考慮する。

3.1. 書評の文書構造

書評の文書構造は、対象となる書籍に関する情報、書評に関する情報を要素として構成する。書籍については、書籍名、著者、出版社、出版年、価格、ISBN を必須要素としてこの順序で記述する。共著の場合を考慮して、著者については複数記載できるよう定義する。書評は、書評としての表題、執筆者、本文を必須要素としてこの順序で記述する。

3.2. 各要素の定義

各要素の定義は、異なるウェブサイトでの再利用を考慮し要素ごとに分割

する。定義する要素名の重複を避けるために名前空間を指定する。名前空間の URI は `http://bookreview/schema` とする。各要素の定義は、図 1 に示す schema 要素の子要素として定義される。なお、以後の説明では、schema 要素の子要素の部分のみ記載する。

```
<?xml version="1.0"?>
<xsd:schema
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns="http://bookreview/shcema"
  targetNamespace="http://bookreview/schema">
  (ここに各要素の定義を記載する)
</xsd:schema>
```

図 1 スキーマ定義のルート要素部分

3. 2. 1. 書籍名・書評の表題

文字列で書籍名、または書評の表題を記述する（図 2 stdTitle.xsd）。副題が添えられる場合を考慮し、0 個以上 1 個以下の副題を定義する。

```
<xsd:element name="title" type="titleType" />
<xsd:complexType name="titleType">
  <xsd:sequence>
    <xsd:element ref="mainTitle" />
    <xsd:element ref="subTitle" minOccurs="0" />
  </xsd:sequence>
</xsd:complexType>
<xsd:element name="mainTitle" type="xsd:string" />
<xsd:element name="subTitle" type="xsd:string" />
```

図 2 title 要素の定義 stdTitle.xsd

3. 2. 2. 著者・執筆者

文字列で著者および執筆者の氏名を姓と名に分けて記述する（図 3 stdAuthor.xsd）。

```
<xsd:element name="author" type="authorType" />
<xsd:complexType name="authorType">
  <xsd:sequence>
    <xsd:element name="firstName" type="xsd:string" />
    <xsd:element name="lastName" type="xsd:string" />
  </xsd:sequence>
</xsd:complexType>
```

図 3 author 要素の定義 stdAuthor.xsd

3. 2. 3. 出版社・出版年・価格・ISBN

出版社名・出版年・価格・ISBN はいずれも文字列で記述する。出版年は西暦で記載する。価格は非負の整数値である。ISBN は 10 桁表示かまたは 13 桁表示のいずれか一方を記載する。これら出版に関する情報には、子要素を定義しない（図 4 stdPublish.xsd）。

```
<xsd:element name="publisher" type="xsd:string" />
<xsd:element name="publishYear" type="xsd:gYear" />
<xsd:element name="price" type="xsd:nonNegativeInteger" />
<xsd:element name="isbn" type="xsd:string" />
```

図 4 出版に関連する要素 stdPublish.xsd

3. 2. 4. 書評本文

書評本文は 1 個以上の段落の繰り返しとして定義する。各段落は文字列で記述する（図 5 stdBody.xsd）。


```
<xsd:element name="body" type="bodyType" />
<xsd:complexType name="bodyType" />
  <xsd:sequence>
    <xsd:element ref="paragraph" maxOccurs="unbounded" />
  </xsd:sequence>
</xsd:complexType>
<xsd:element name="paragraph" type="xsd:string" />
```

図 5 body 要素の定義 stdBody.xsd

3. 3. 書籍に関する情報

各要素の定義を参照し、書籍に関する情報を記述するための文書構造を定義する(図 6 (1)・(2) stdBook.xsd)。参照される各要素の定義は、同一の名前空間にあるものとする。

```
<xsd:include schemaLocation="stdTitle.xsd" />
<xsd:include schemaLocation="stdAuthor.xsd" />
<xsd:include schemaLocation="stdPublish.xsd" />

<xsd:element name="book" type="bookType" />
<xsd:complexType name="bookType">
  <xsd:sequence>
    <xsd:element ref="title" />
    <xsd:element ref="author" maxOccurs="unbounded" />
    <xsd:element ref="publisher" />
    <xsd:element ref="publishYear" />
    <xsd:element ref="price" />
```

図 6 (1) book 要素の定義 stdBook.xsd

```
<xsd:element ref="isbn" />
</xsd:sequence>
</xsd:complexType>
```

図 6 (2) book 要素の定義 stdBook.xsd

3. 4. 書評に関する情報

書籍に関する情報と同様に、各要素の定義を参照し、書評に関する情報を記述する文書構造を定義する（図 7 stdReview.xsd）。

```
<xsd:element name="review" type="reviewType" />
<xsd:complexType name="reviewType">
  <xsd:sequence>
    <xsd:element ref="title" />
    <xsd:element ref="author" />
    <xsd:element ref="body" />
  </xsd:sequence>
</xsd:complexType>
```

図 7 review 要素の定義 stdReview.xsd

3. 5. 書評用スキーマ

以上から、書評用スキーマを次のように定義する（図 8 stdBReview.xsd）。各要素の定義を include 要素により埋め込み、book 要素と review 要素を子要素とする bookReview 要素を定義する。そののち、book 要素の子要素と review 要素の子要素を定義したものである（なお、以後のスキーマの記述は、紙面の都合上、やや文字を小さくし、行間を狭めている）。

```
<?xml version="1.0"?>
<xsd:schema
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns="http://bookreview/shcema"
  targetNamespace="http://bookreview/schema">

  <xsd:include schemaLocation="stdTitle.xsd" />
  <xsd:include schemaLocation="stdAuthor.xsd" />
  <xsd:include schemaLocation="stdPublish.xsd" />
  <xsd:include schemaLocation="stdBody.xsd" />

  <xsd:element name="bookReview" type="bookReviewType" />
  <xsd:complexType name="bookReviewType">
    <xsd:sequence>
      <xsd:element ref="book" />
      <xsd:element ref="review" />
    </xsd:sequence>
  </xsd:complexType>

  <xsd:element name="book" type="bookType" />
  <xsd:complexType name="bookType">
    <xsd:sequence>
      <xsd:element ref="title" />
      <xsd:element ref="author" maxOccurs="unbounded" />
      <xsd:element ref="publisher" />
      <xsd:element ref="publishYear" />
      <xsd:element ref="price" />
      <xsd:element ref="isbn" />
    </xsd:sequence>
  </xsd:complexType>

  <xsd:element name="review" type="reviewType" />
  <xsd:complexType name="reviewType">
    <xsd:sequence>
      <xsd:element ref="title" />
      <xsd:element ref="author" />
      <xsd:element ref="body" />
    </xsd:sequence>
  </xsd:complexType>

</xsd:schema>
```

図 8 書評用スキーマ stdBReview.xsd

4. 書評サイトへの適用

定義された書評用スキーマに対して各書評サイトが独自の拡張を施し、それぞれの書評サイトで利用する。ここでは、和歌山大学の書評サイト「リトルネロ」を対象として拡張内容を検討する。

4.1. ウェブサイト独自の拡張

リトルネロで公開する書評は、学内に所属する者により執筆されている。公開時には執筆者の所属と学生・職員・教員の区別を表示している。これらを記述するための定義をリトルネロ側に追加する必要がある。また、リトルネロでは、書籍に加え音楽と映画についての論評も掲載している。音楽と映画の作品についての情報を記載する文書構造の定義が必要である。いずれも、既に定義されたスキーマを再利用して定義する。

4.2. 音楽に関する情報

音楽に関しては、作品名、演奏者、作曲者、作詞者、レーベル、発表年、価格を記載する。作品名は `titleType`、演奏者、作曲者、作詞者は `authorType`、レーベルは `publisher` 要素、発表年は `publishYear` 要素、価格は `price` 要素を再利用して定義する。

4.3. 映画に関する情報

映画に関しては、作品名、主演、助演、監督、配給会社、発表年を記載する。作品名は `titleType`、主演俳優、助演俳優、監督は `authorType`、配給会社は `publisher` 要素、発表年は `publishYear` 要素を再利用して定義する。

4.4. 執筆者に関する追加情報

執筆者については、所属と職種（学生も含むものとする）を追加記載する。これらは新たに定義する。

4.5. リトルネロ用スキーマ

図8に示した書評用スキーマの名前空間接頭辞を `br`、リトルネロの名前空間 URI を `http://mysite/schema/` として、リトルネロ用スキーマを定義する(図

9 (1) ~ (3) ritornell.xsd。

名前空間 `br=http://bookreview/schema` の各要素定義を `import` 要素を用いて取り込む。書籍情報用の `br:bookReview` 要素、音楽情報用に新たに定義する `musicReview` 要素、映画情報用に新たに定義する `movieReview` 要素を子要素として持つ `ritornello` 要素を定義する。`musicReview` 要素は、子要素として `br:title`, `br:publisher`, `br:publishYear`, `br:price` の各要素を再利用し、新たに `player`, `composer`, `songwriter` の各要素を子要素として定義する。`movieReview` 要素は子要素として `br:title`, `br:publisher`, `br:publishYear` の各要素を再利用し、`lead`, `support`, `director` の各要素を子要素として定義する。書評情報に相当する `ritornelloReview` 要素は、子要素として `br:title`, `br:body` の各要素を再利用し、`br:authType` を拡張して `reviewer` 要素を子要素として定義する。

```
<?xml version="1.0"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:br="http://bookreview/schema"
  xmlns="http://mysite/schema"
  targetNamespace="http://mysite/schema/">

  <xsd:import
    namespace="http://bookreview/schema"
    schemaLocation="http://bookreview/schema/stdBook.xsd" />
  <xsd:import
    namespace="http://bookreview/schema"
    schemaLocation="http://bookreview/schema/stdBody.xsd" />

  <xsd:element name="ritornello" type="ritornelloType" />
  <xsd:complexType name="ritornelloType">
    <xsd:choice>
      <xsd:element ref="br:bookReview" />
      <xsd:element ref="musicReview" />
      <xsd:element ref="movieReview" />
    </xsd:choice>
  </xsd:complexType>
```

図 9 (1) リトルネロ用スキーマ ritornello.xsd

```
<xsd:element name="musicReview" type="musicReviewType" />
<xsd:complexType name="musicReviewType" >
  <xsd:sequence>
    <xsd:element ref="br:title" />
    <xsd:element name="player" type="br:authorType" />
    <xsd:element name="composer" type="br:authorType" />
    <xsd:element name="songwriter" type="br:authorType" />
    <xsd:element ref="br:publisher" />
    <xsd:element ref="br:publishYear" />
    <xsd:element ref="br:price" />
  </xsd:sequence>
</xsd:complexType>

<xsd:element name="movieReview" type="movieReviewType" />
<xsd:complexType name="movieReviewType" >
  <xsd:sequence>
    <xsd:element ref="br:title" />
    <xsd:element name="lead" type="br:authorType" />
    <xsd:element name="support" type="br:authorType" />
    <xsd:element name="director" type="br:authorType" />
    <xsd:element ref="br:publisher" />
    <xsd:element ref="br:publishYear" />
  </xsd:sequence>
</xsd:complexType>

<xsd:element name="ritornelloReview" type="ritornelloReviewType" />
<xsd:complexType name="ritornelloReviewType">
  <xsd:sequence>
    <xsd:element ref="br:title" />
    <xsd:element ref="reviewer" />
    <xsd:element ref="br:body" />
  </xsd:sequence>
</xsd:complexType>

<xsd:element name="reviewer" type="reviewerType" />
<xsd:complexType="reviewerType">
  <xsd:complexContent>
    <xsd:extension base="br:authType">
    <xsd:sequence>
      <xsd:element ref="department" />
      <xsd:element ref="occupation" />
    </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

図 9 (2) リトルネロ用スキーマ ritornello.xsd

```
</xsd:sequence>
</xsd:complexContent>
</xsd:complexType>
<xsd:element name="department" type="xsd:string" />
<xsd:element name="occupation" type="xsd:string" />
</xsd:schema>
```

図 9 (3) リトルネロ用スキーマ ritornello.xsd

5. おわりに

ウェブサイト間での書評の相互提供を実現するためのアプローチとして、書評を XML 文書として記述することを提案し、その文書構造を定める書評用スキーマを XML Schema により定義した。また、書評を集約するウェブサイトの類型としては、マッシュアップ型サイトに優位性があることを述べた。書評用スキーマは、必要に応じて他のウェブサイトで独自拡張を施せるように、複合型の要素ごとにスキーマ文書を作成している。

提案した書評用スキーマを利用して、和歌山大学の書評サイト用スキーマを構成した。実際のウェブサイトでは、現在 XML 文書の形式を統一しているもののスキーマは定義していない。和歌山大学の書評サイト用スキーマを定めたことにより、他のウェブサイトへの書評提供や、携帯端末による閲覧に適した HTML 文書への変換プログラムを実装する技術的な準備が整いつつある。

今後、和歌山大学の書評サイト用スキーマを実際のウェブサイトに適用し、既存文書の構造との整合性を検証する必要がある。これに加え HTML 文書への変換プログラムの修正、携帯端末からの閲覧への対応、外部提供のための API の実装などが今後の課題である。